What is claimed is:

1. A central processing unit (CPU) for easily testing and debugging an application program, including a universal register file for temporarily storing data necessary for operation of data and address, a program counter storing addresses at which programs are stored, a special register file having a status register indicating a status of the CPU and a break register, an internal bus connecting the universal register file and the special register file, and a control unit connected to the internal bus, for outputting various control signals necessary for internal and external components of the CPU, the CPU comprising:

a data communications unit for performing data communications with a host computer;

a status register having a flag representing whether an operational mode of the CPU is a general operational mode representing a general operational state or a debugging mode representing a debugging state;

a debugging stack pointer register which is used as a stack pointer designating a stack memory storing data of a debugging program; and

a comparator for comparing a value stored in a break register with break data,

wherein the CPU is converted into the debugging mode if

the break register value is same as the break data, the flag of the status register has a value representing a debugging mode, a start address for performing a debugging program is loaded in a program counter, and the debugging program is executed to perform a debugging according to a command from the host computer via the data communications unit.

2. The CPU of claim 1, further comprising:

a reset data storage unit storing reset data; and

a reset data comparator for comparing the data input via the data communications unit with the reset data stored in the reset data storage unit, and instructing the control unit to initialize the CPU if the data input via the data communications is same as the reset data.

3. The CPU of claim 1, wherein said break data is a program address stored in the program counter.

4. The CPU of claim 1, further comprising a mask register, wherein the break data is a result of operation of the values stored in the program counter and the mask register.

5. The CPU of claim 1, wherein the break data is a memory address at which data is stored.

6. The CPU of claim 1, further comprising a mask register, wherein the break data is a result of operation of a memory address at which data is stored and the value stored in the mask register.

7. The CPU of claim 1, wherein said break data is data input to and output from the CPU.

8. The CPU of claim 1, further comprising a mask register, wherein the break data is a result of operation of data input to and output from the CPU and the value stored in the mask register.

9. The CPU of claim 1, wherein said break data is an address input to and output from the CPU.

10. The CPU of claim 1, further comprising a mask register, wherein the break data is a result of operation of an address input to and output from the CPU and the value stored in the mask register.

11. The CPU of claim 1, wherein said control unit receives a debugging memory select signal and loads a respectively different address for performing a debugging program in the program counter according to the debugging

memory select signal when the value stored in the break register is same as the break data.

12. The CPU of claim 1, further comprising a data storage memory storing data values used for a debugging program, which is separated from a data storage memory storing data values used for a general program.

13. The CPU of claim 1, further comprising a memory storing a debugging program, which is separated from a memory storing a general program.

14. The CPU of claim 1, wherein an application program to be tested and debugged is downloaded from the host computer via the data communications unit.

15. The CPU of claim 1, wherein the value stored in the program counter and the data stored in the status register are stored in a memory designated by the debugging stack pointer register, when the CPU has been converted into a debugging mode.

16. The CPU of claim 1, further comprising a temporary storage register, wherein the value stored in the program counter and the data stored in the status

register are stored in a temporary storage memory when the CPU has been converted into a debugging mode.

17. The CPU of claim 1, further comprising:

a reference data storage unit storing reference data; and

a reference data comparator for comparing the data input via the data communications unit with the reference data,

wherein the control unit controls the CPU to be converted into a debugging mode, and loads a start address for performing a debugging program in the program counter, if the data input via the data communications is same as the reference data, to thereby control the CPU to perform a debugging according to a command from the host computer via the data communications unit.

18. A central processing unit (CPU) for easily testing and debugging a program, including a universal register file for temporarily storing data necessary for operation of data and address, a program counter storing addresses at which programs are stored, a special register file having a status register indicating a status of the CPU and a break register, and an internal bus connecting the universal register file and the

special register file, the CPU comprising:

a data communications unit for performing data communications with a host computer;

a status register having a flag representing whether an operational mode of the CPU is a general operational mode representing a general operational state or a debugging mode representing a debugging state;

a debugging stack pointer register designating a stack memory storing data of a debugging initialization program and data of a debugging service program;

a control unit for initializing the CPU by a rest signal, checking a debugging mode proceeding signal, loading a start address for performing a debugging initializing program in a program counter if the debugging mode proceeding signal has been activated, to thereby converting the CPU into the debugging initialization mode, and setting the flag of the status register into a value representing the debugging mode, and outputting various control signals necessary for internal and external components of the CPU connected to an internal bus; and

a comparator for comparing a value stored in a break register with break data,

wherein the CPU is converted into the debugging mode if the break register value is same as the break data,

the flag of the status register has a value representing a debugging service mode, a start address for performing a debugging service program is loaded in a program counter, and the debugging service program is executed to perform a debugging according to a command from the host computer via the data communications unit.

19. The CPU of claim 18, further comprising:

a reset data storage unit storing reset data; and

a reset data comparator for comparing the data input via the data communications unit with the reset data stored in the reset data storage unit, and instructing the control unit to initialize the CPU if the data input via the data communications is same as the reset data.

20. The CPU of claim 18, wherein said break data is a program address stored in the program counter.

21. The CPU of claim 18, further comprising a mask register, wherein the break data is a result of operation of the values stored in the program counter and the mask register.

22. The CPU of claim 18, wherein the break data is a memory address at which data is stored.

23. The CPU of claim 18, further comprising a mask register, wherein the break data is a result of operation of a memory address at which data is stored and the value stored in the mask register.

24. The CPU of claim 18, wherein said break data is data input to and output from the CPU.

25. The CPU of claim 18, further comprising a mask register, wherein the break data is a result of operation of data input to and output from the CPU and the value stored in the mask register.

26. The CPU of claim 18, wherein said break data is an address input to and output from the CPU.

27. The CPU of claim 18, further comprising a mask register, wherein the break data is a result of operation of an address input to and output from the CPU and the value stored in the mask register.

28. The CPU of claim 18, wherein said control unit receives a debugging memory select signal and loads a respectively different address for performing a debugging program in the program counter according to the debugging

memory select signal when the value stored in the break register is same as the break data.

29. The CPU of claim 18, further comprising a data storage memory storing data values used for a debugging program, which is separated from a data storage memory storing data values used for a general program.

30. The CPU of claim 18, further comprising a memory storing a debugging program, which is separated from a memory storing a general program.

31. The CPU of claim 18, wherein an application program to be tested and debugged is downloaded from the host computer via the data communications unit.

32. The CPU of claim 18, further comprising a temporary storage register, wherein the value stored in the program counter and the data stored in the status register are stored in a temporary storage memory when the CPU has been converted into a debugging mode.

33. The CPU of claim 18, further comprising:

a reference data storage unit storing reference data; and

a reference data comparator for comparing the data input via the data communications unit with the reference data,

wherein the control unit controls the CPU to be converted into a debugging mode, and loads a start address for performing a debugging program in the program counter, if the data input via the data communications is same as the reference data, to thereby control the CPU to perform a debugging according to a command from the host computer via the data communications unit.